

NO. 5:10-CV-25-FL

Defendant.

BACKGROUND

As relevant to the instant motion, the court summarizes the following facts. Plaintiff and defendant are competing software companies. In or around 2003, defendant obtained from

Case 5:10-cv-00025-FL Document 556 Filed 11/05/15 Page 1 of 18

Amazon.com several copies of SAS Learning Edition (“SAS LE”), a statistical software product created and sold by plaintiff and intended assist users in learning the “language of SAS,” a computer language free for anyone to use and in which various programs for data management and analysis may be written. At the time defendant’s employees installed SAS LE, defendant, through those employees, entered into the SAS LE license agreement, which restricted use of that software to “non-production purposes only” and prohibited “reverse engineering.” That license agreement is at the heart of this case.

Defendant used SAS LE to assist in the development of a competing software product, the World Programming System (“WPS”). Defendant’s owners intended WPS to replace the SAS System, a comprehensive set of computer software products created and distributed by plaintiff, similar in nature to SAS LE but with greater data processing capabilities. Prior to defendant first obtaining SAS LE, its owners and employees had been developing WPS using only SAS “manuals,” online documentation addressing the SAS System’s functionality. After defendant obtained SAS LE, its employees compared that software’s output with the output of the developing WPS in an effort to have WPS’s output mimic SAS LE’s, and ultimately the SAS System’s functionality. Specifically, and among other things, defendant’s employees would input code into SAS LE, observe its output, and modify the WPS source code in an effort to achieve a one-to-one input-output correspondence.

Plaintiff filed suit in January 2010. As relevant here, plaintiff asserted a breach of contract claim, alleging defendant’s use of SAS LE violated the license agreement’s restriction on use to “non-production purposes” and “reverse engineering” prohibition. The court granted summary judgment in favor of plaintiff on both grounds. See SAS I, 64 F. Supp. 3d at 769–774, as modified

SAS Inst., Inc. v. World Programming Ltd. (“SAS III”), 5:10-CV-25-FL, slip op., at *9–19 (E.D.N.C. July 21, 2015), and as supplemented SAS Inst., Inc. v. World Programming Ltd. (“SAS II”), 5:10-CV-25-FL, slip op., at *6–13 (E.D.N.C. July 1, 2015). The case went to trial on the issue of damages on September 22, 2015, with defendant contending plaintiff suffered only nominal damages from defendant’s breach of the license agreement.² Prior to trial, the parties agreed that any damages calculation necessarily depended on the resolution of a hypothetical question: whether it would have been possible for defendant to develop WPS without use of SAS LE, and, if so, how much additional development time would have been needed. To that end, the parties each retained experts. Plaintiff’s expert, Dr. James A. Storer, opined that defendant could not have developed WPS without use of SAS LE. See SAS IV, 2015 WL 4757397, at *3–4 (summarizing Storer’s opinion).³ Defendant’s expert, Roman, offered two opinions. First, he opined generally that WPS could have been developed without use of SAS LE, which portion the court allowed to be presented at trial. Second, he offered the opinion, which was excluded from trial based on the instant motion, that any delay occasioned by defendant’s loss of SAS LE, at most, would be 19 days (the “cumulative delay” calculation).

² The case culminated in a three-week trial lasting from September 22, 2015, until October 9, 2015. At the conclusion of trial, the jury returned a verdict in favor of plaintiff, finding plaintiff had been actually damaged by defendant’s breach of contract in the amount of \$26,376,635.00. The jury also returned a verdicts in plaintiff’s favor for the same amount on two other causes of action, fraud and violation of the North Carolina Unfair Trade Practices Act, N.C. Gen. Stat. § 75–1.1 et seq.. In addition, the jury awarded plaintiff \$3,000,000.00 in punitive damages on its successful fraud claim. Several motions related to the trial remain pending. (See, e.g., DE 464, 482, 505). The court will address these motions by separate order to follow.

³ On May 4, 2015, defendant filed a motion to exclude Storer’s testimony in its entirety under Federal Rule of Evidence 702, or, in the alternative, to exclude certain portions of Storer’s testimony under Federal Rule of Evidence 403. On August 12, 2015, the court entered order granting in part and denying in part defendant’s motion. In particular, the court denied defendant’s motion to the extent it sought to exclude Storer’s testimony under Rule 702. However, the court granted in part defendant’s motion to exclude certain portions of Storer’s testimony under Rule 403, concluding that Storer’s testimony was unnecessarily confusing inasmuch as he opined the SAS System was a “a system of interpreters and compilers,” rather than “an interpreter or compiler,” as was found by the court in its order on summary judgment. See generally, SAS IV, 2015 WL 4757397, at *10.

In its motion to exclude Roman's cumulative delay calculation, plaintiff argued Roman's opinion was speculative, based on insufficient evidence of record, and was not grounded in Roman's experience, education, or training. In particular, plaintiff contended that Roman was unable to point to any particular academic source or past experience that supported his cumulative delay calculation, but, rather, merely assumed that such calculation was correct. At hearing held July 23 and 24, 2015, the court granted plaintiff's motion. In its oral ruling, the court relied on General Electric v. Joiner, 522 U.S. 136 (1997), and held that Roman's cumulative delay opinion was too speculative to be admitted at trial.

COURT'S DISCUSSION

A. Standard of Review

The admission of expert testimony is governed by Rule 702 of the Federal Rules of Evidence. The proponent of the expert testimony bears the burden of establishing its admissibility by a preponderance of proof. Cooper v. Smith & Nephew, Inc., 259 F.3d 194, 199 (4th Cir. 2001). A district court is granted broad latitude in making its determination on the admissibility of proposed expert testimony. United States v. Gastiburo, 16 F.3d 582, 589 (4th Cir. 1994) ("The trial judge has broad discretion under Rule 702."). Review by the advisory committee shows that the rejection of expert testimony is the exception rather than the rule. Fed. R. Evid. 702 advisory committee's note (2000).

Rule 702 provides that expert testimony is appropriate when it "will assist the trier of fact to understand the evidence or to determine a fact in issue." Fed. R. Evid. 702. Rule 702 further provides that a witness qualified as an expert may be permitted to testify where "(1) the testimony is based upon sufficient facts or data, (2) the testimony is the product of reliable principles and

methods, and (3) the witness has applied the principles and methods reliably to the facts of the case.” Id. Courts have distilled the requirements of Rule 702 into two crucial inquiries: whether the proposed expert’s testimony is relevant and whether it is reliable. Kumho Tire Co. v. Carmichael, 526 U.S. 137, 141 (1999); Daubert, 509 U.S. at 589; United States v. Forrest, 429 F.3d 73, 80 (4th Cir. 2005). The trial court must carry out the special gate-keeping obligation of ensuring that expert testimony meets both requirements. Kumho Tire, 526 U.S. at 147; United States v. Moreland, 437 F.3d 424, 431 (4th Cir. 2006), overruling on other grounds recognized by United States v. Diosdado-Star, 630 F.3d 359 (4th Cir. 2011).

In order to be considered relevant, the proposed expert testimony must appear to be helpful to the trier of fact. See Daubert, 509 U.S. at 591-92. “Testimony from an expert is presumed to be helpful unless it concerns matters within the everyday knowledge and experience of a lay juror.” Kopf v. Skyrn, 993 F.2d 374, 377 (4th Cir. 1993).

“‘[T]he test of reliability is flexible’ and ‘the law grants a district court the same broad latitude when it decides how to determine reliability as it enjoys in respect to its ultimate reliability determination.’” United States v. Wilson, 484 F. 3d 267, 274 (4th Cir. 2007) (quoting Kumho Tire, 526 U.S. at 141-42).

One factor pertinent to reliability is the proposed expert’s qualifications. See Giddings v. Bristol-Myers Squibb Co., 192 F.Supp.2d 421, 425 (D.Md. 2002). A witness may qualify to render expert opinions in any one of the five ways listed in Rule 702: knowledge, skill, experience, training, or education. Kumho Tire, 526 U.S. at 147. The Fourth Circuit has ruled that when an expert’s qualifications are challenged, “‘the test for exclusion is a strict one, and the purported expert must have neither satisfactory knowledge, skill, experience, training nor education on the

issue for which the opinion is proffered.” Kopf, 993 F.2d at 377 (quoting Thomas J Kline, Inc. v. Lorillard, Inc., 878 F.2d 791, 799 (4th Cir. 1989)).

Additional factors also bear on the reliability of the expert’s testimony. They may include: “(1) whether a theory or technique can be (and has been) tested; (2) whether the theory or technique has been subjected to peer review and publication; (3) whether a technique has a high known or potential rate of error and whether there are standards controlling its application; and (4) whether the theory or technique enjoys general acceptance within the relevant community.” Tunnell v. Ford Motor Co., 245 F. App’x. 283, 286 (4th Cir. 2007) (citing Kumho Tire, 526 U.S. at 149-50); accord Daubert, 509 U.S. at 593-94. In addition, “[a]n expert’s opinion should be excluded when it is based on assumptions which are speculative and are not supported by the record.” Tyger Const. Co. Inc. v. Pensacola Const. Co., 29 F.3d 137, 142 (4th Cir. 1994).

B. Summary of Roman’s Qualifications and Opinions

Roman is the owner, founder, and CEO of MediaFrame, Inc. (“MediaFrame”), located in Santa Clara, California. MediaFrame does practical research and development, as well as litigation consulting for a variety of law firms, principally in the area of patent law. He holds a bachelor’s degree in Computer Science from Brigham Young University in Provo, Utah. In addition to his academic credentials, Roman has two publications in the area of software programming and holds many issued patents. Roman has extensive experience in the area of computer software development and has past experience working collaboratively with teams to develop commercial software products. His credentials have garnered him respect from federal courts in the past. In particular, he has been both a court appointed expert witness, as well as a special master.

As pertinent to the instant motion, Roman was to testify that the development of WPS would have taken 19 additional days if defendant's developers had been denied use of SAS LE. Roman believes that any hypothetical delay created by loss of use of SAS LE would have resulted from the development method that then would have been required. In particular, because defendant would have needed to send test programs to third parties that owned SAS System licenses in order to generate test results, some delay would have accumulated given that defendant's programmers then would have needed to shift their focus to a new task while waiting for those results and then shift their focus back to the previous task once the results were received. This hypothesis accounts for a brief work stoppage when switching between tasks and time spent becoming reacquainted with the work after receiving the test results. Roman based his opinion in prior research on the importance of sub-second response times, which posits that mental efficiencies are lost when it takes one second or longer to receive a response to a computer-based inquiry. To arrive at his 19-day figure, Roman examined the number of new "product source code" files added to WPS between releases of new versions of the WPS software.⁴ He then assumed that, using three programmers, every 100 new files delayed production by one day. Using that calculation, Roman then determined whether the development delay, if any, would have impacted the release date for a given version of WPS.

C. Analysis

⁴ A "product source code file" is "the source code files that actually go into the object code that then gets shipped to the consumer. [By contrast,] . . . test source code files and . . . test data files, those are part of th[e] test scaffold." (Roman Dep., DE 384, 260:16–21).

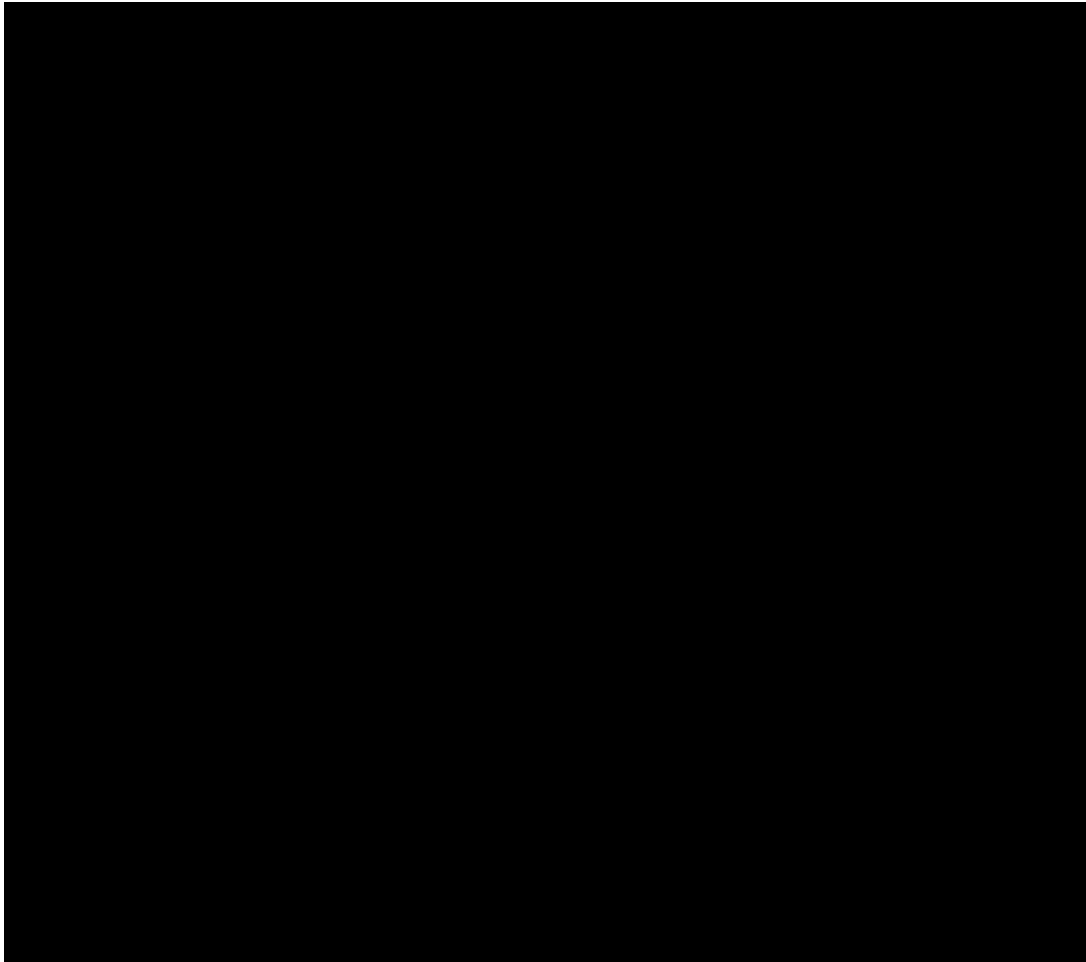
A reliable expert opinion grounded in that expert's experience, training, or education must provide sufficient detail to demonstrate how the expert's background leads to the conclusion reached. SAS IV, 2015 WL 4757397, at *7. In addition, "[a] district court may conclude that there is simply too great an analytical gap between the data and the opinion proffered." Gen. Elec. Corp. v. Joiner, 522 U.S. 136, 146 (1997). When such a gap exists, the court may exclude the expert's proffered opinion on that basis. See, e.g., Tyger, 29 F.3d at 142 ("An expert's opinion should be excluded when it is based on assumptions which are speculative."); see also EEOC v. Freeman, 778 F.3d 463, 472 (4th Cir. 2015) (Agee, J., concurring).

Roman failed to meaningfully explain how the number of new product source code files in the WPS software, as it exists today, reasonably could approximate the delay that would be incurred had defendant's programmers lost use of SAS LE. In his report, Roman opines:


In my opinion, the cumulative delay to getting WPS to market if WPL had not used the SAS Learning Edition would have been 19 days. My analysis of the 48 Issues identified in Roman Exhibit 5 shows that none of them would have contributed to any delay in any release of any version. Several were in fact put off several times until a later version. Others were fixed in less than an hour or two or within few days of being reported. As discussed throughout this report, WPL could have gotten the results of the SAS Learning Edition in other equally efficient ways. However, it is arguable that there would have been some cumulative increase in programmer efficiency by being able to learn about the nuances of the language of SAS and particular statements or outputs by being able to use the SAS Learning Edition on their desktop PC rather than sending it to be run on a remote computer, such as a customer's mainframe. I allocated one day delay for each 100 new product source code files as shown in Roman Exhibit 8-6 and detailed in Roman Exhibit 8-8.

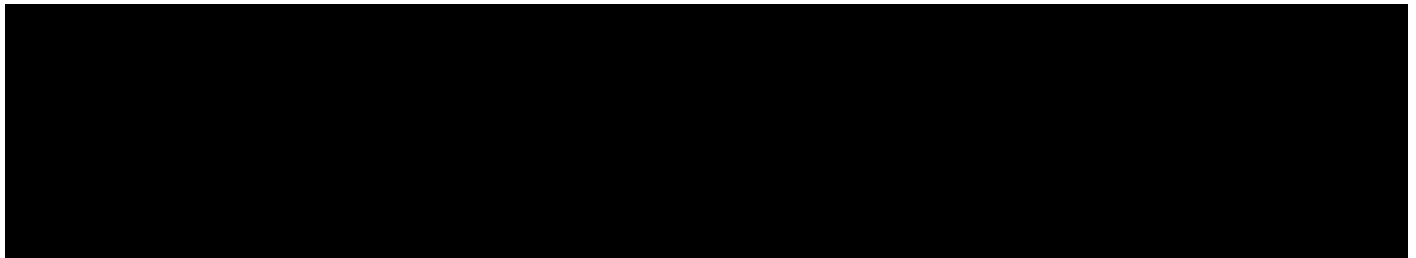
(Roman Report, DE 350, ¶204). Roman Exhibit 8-8, referenced above lists the various versions of WPS, the total number of source files included in each version, the total number of lines of code included in those source files, the number of new source files added, and number of new lines of code included in those new source files, among other things. As relevant here, that chart provides:

Verison	WPS Product Source Files	Prior Feature Source Files	New Feature Source Files
---------	--------------------------	----------------------------	--------------------------



(Roman Ex. 8, DE 350-10, 9).

This table represents only a small sample of the various updates to the WPS software defendant released. Thus, there is not a perfect correspondence between the “WPS Product Source Files” in a listed a particular version and the “Prior Feature Source Files” in the next listed version. For example, version 2.3 is listed as having  product source files while version 2.4 lists the



prior version as having [REDACTED] product source files. This is because between versions 2.3 and 2.4, defendant released WPS versions 2.3.1, 2.3.2, 2.3.3, 2.3.4, and 2.3.5. (See Roman Ex. 9, DE 385-11). Each of these versions increased the total number of new product source code files. Roman included in his analysis only “on versions that had significant amounts of change or where [he] identified specific new products and features being implemented.” (Roman Report, DE 350, ¶147).

Roman determined that the addition of the “New Feature Source Files” would have resulted in a market delay in only seven instances, corresponding to versions 2.1.0, 2.2.0, 2.3.0, 2.4.0, 2.5, 2.5.1, and 2.5.1. (See Roman Report ¶205). These versions would have been delayed 4, 1, 5, 3, 4, 1, and 1 day(s), respectively. (Id.). Although Roman’s 19 day calculation follows from that analysis, his report provides no meaningful guidance to assist in understanding why Roman attributed one day of delay to every 100 new source code files.

At his deposition, Roman failed to clarify his assumption that the release of WPS would have been delayed one day for every 100 new source code files included in a given version. Rather, he generated a substantial amount of confusion. He explained his opinion as follows:

[A.] So then I looked at the different versions, and I did an analysis of the new - - the new features in each product, and, again, those are shown in Exhibit [8-6] and in [8-8]. And in paragraph 205, I basically have a summary of each version, and I allocated the delay in these versions based on the amount of new source code that was developed. And, in particular, on the top of page 47, I allocated one-day delay for each 100 new product source code files. And when those were added up, the answer was 19.

(Roman Dep., DE 384, 255:8–19). He then went on to explain:

[A.] I basically said it could be more efficient [to send test files to third parties], but I’m going to assume it was equally efficient to send it to a third party and have the third party run it and give you the results.

Q. Well, if you assumed that it’s equally efficient, why are there any delays at all?

- A. I looked at - - there was [sic] a large number of new files being developed, and - - and in the early days of my career, for example, at IBM, we did a lot of - - and also at Dialogic, we did a - - a lot of research into the value of having subsecond response time. And so one of the goals at Dialogic was for - - to have each of our commands finish in less than a second.

And, basically, the studies show that if a single action took more than a second, then the user would get distracted. You know, they might have a sip of coffee or - - or start thinking about a different problem, and, therefore, their overall efficiency would decline.

And so, taking all that into context, I basically said, well, it's arguable that even if the - - the bulk processing of a group of things in this process that's disclosed in the evidence maybe was more efficient, but the fact that you're handing off someone else [sic] and then getting the results back, you have to change your focus. You're not getting subsecond response.

And therefore, I said it's arguable that even though the actual running and the cumulative time to run a batch of things through the mainframe might be faster, it could be argued that there's some inefficiency due to the fact that you're handing off to someone else and then have to kind of come back to it.

And so based on that general understanding about subsecond response time and - - and the general argument that could be made that doing it on your desktop is more efficient than sending it off to a batch process - - . . . then I said, "Okay. I think it's fair to assume that there's some delay."

(Roman Dep. 257:16–259:8). When asked at his deposition about the basis for his "one day delay for each 100 new product source code files" opinion, Roman stated:

- A. Well, so going back to - - you know, it's possible that there was no delay if you assume that the efficiency was the same. But I think it's arguable, for the reasons I already stated at length, that there was some delay or that there would have been some delay, and so - - so, for example, this [REDACTED]

So let's say that these three procedures were developed by three different programmers, and they all - - well, let's say it was in 33 files each. So that's a hundred, that those three programmers each were responsible for a new procedure that comprised 33 files.

...

So over the course of the development of that - - those three procedures of which equate to, in my hypothetical, a hundred files, say, over the course of that development release, the inefficiency would be this "I lose five second here or five seconds there because I had to change gears," essentially. And those cumulative five seconds here and there or few minutes extra to run it through an alternative source would add up to be one day for that programmer working on a feature.

Q. Okay.

A. So they might spend two weeks working on a particular feature, so it would just be a few minutes each day cumulated to one programmer a day.

...

Q. Okay. I'm - - I'm trying to understand, is there anywhere in the record that you've seen that would provide data that would support the one-day delay per 100 new product source code files where we can make a calculation, in other words?

...

[A.] I think essentially what we have is my analysis, and that's based on my experience in doing software analysis and software development methodology and subsecond response time research throughout my career.

Q. Do you have any experience where, for example, you were - - instead of being required to use a tool that was available on your desktop, you had to have it processed remotely? Is that something that - - that you had as - - as part of your experience?

A. Yeah. That's exactly what the whole Dialogic product was involved in.

Q. Okay.

A. Dialogic's customers were programmers who use IBM mainframes, and when they made a change to their program, they had to submit it to the mainframe to be compiled, and Dialogic's project was an editor which could do the syntax checking on a UNIX-based front-end processor. And so - -

Q. Okay. My question is in the context of development.

Instead of having the tool that's available to you in the context of developing some software, having it available to you on a laptop, you're having to process it remotely.

Was that something you experienced?

A. Right. And, again, that's exactly what Dialogic did.

Q. Okay.

A. Our customers were developers in insurance companies and banks - - . . . and they were developing software and - - well, the - - the reason they bought us is because we gave them better programmer performance because we had subsecond response time.

Q. Okay. And was there any - - was there any quantification of the delays encountered by doing it one way versus another way in that context, that you're aware of?

A. Yeah. So as I said, it was the main - - the main - - the main marketing kind of purpose of Dialogic, the reason why people bought the Dialogic machine was to improve their programmer productivity. And - - and so the research we did, the stuff we used in our marketing papers is all based on kind of empirical analysis of the impact of subsecond response time.

Q. None - - none of that's in this record, though; right?

A. No.

Q. And none of it - - none of it's cited in your report; right?

A. That's correct.

Q. Okay.

A. And then, basically, a lot of that research was - - was proprietary and - -

Q. So you couldn't tell us even if you wanted to; right?

A. Yeah. If I - - if I had access, which I did at Dialogic or Dialogic's own information, I don't have access to it today. Yeah.

Q. Okay.

- A. But I do know that there's a large body of work out there that could be obtained - - or at least used to be able to be obtained.
- Q. And, again, one thing that you didn't do was simply inquire of [defendant's programmers] what they thought the delay would have been had they not had Learning Edition and had to submit everything remotely; right?
- A. Yeah. I think I've already answered that. I - - I didn't inquire of them.
- Q. Okay.
- A. Other than communication through counsel.
- Q. Okay. And that wasn't one of the communications that you put to them through counsel right?
- ...
- A. Yeah. So what I will say is I developed this number, and I presented it to counsel. And I requested that they check with [defendant] to see if that was reasonable, and the answer was that the opinions stated in my report were reasonable.

(Roman Dep. 262:17–269:5).

Roman was given a third opportunity to explain his methodology, at a hearing before the court on July 23 and 24, 2015. There, when asked about his attempt to quantify with precision the additional time necessary to develop WPS without use of SAS LE, he stated:

- [A.] I can say for [a] single feature if there were an accumulation of the sub-second delays because of responding to customer feedback at a later time, it would have added up, and again, as I said, it might be five days, it might be 20 days. What I'm saying is that this arguable cumulative delay is a small number, and I'm saying it's arguably greater than zero because of the witness testimony and stuff about this being a long iterative process, and so what I'm trying to do is put a quantified number on it, and again - -
- Q. I understand that you say it's a small number, but my question is, you know, how do you know it's this number, and how many five second delays does it take to get to a one day delay per 100 files?
- A. Well, the one day delay is, you know, divided by - - well, assume 8 hours, and again, that's even a conservative estimate, a lot of programmers don't

work only 8 hours, but 8 hours divided by 60 gives you minutes, divided by another 60 gives you seconds, and then - -

Q. You've actually got to multiply it, right?

A. Yeah. And then - -

Q. So we're talking about thirty-some thousand five-second delays that are somehow creeping into this one day delay per 100? And you don't have a document where you make that calculation, right?

A. No. And again, you know, I think the point you're making is it might be too big, that 30,000 five-second delays may not be justified if you're only doing 29 files or 33 files per programmer. but its not - - it's measured in days, it's not measured in months or years, and again, I'm trying - - you know, there is no hard empirical data, you haven't presented any theory that is anything other than all, and - -

Q. Well, mind you, I'm not the proponent of theories.

A. And "all" is wrong, so I'm trying to say what could you argue, and again, I thought there might be some rebuttal or alternative response and so I was trying to anticipate what you might argue, and again, we have nothing empirical.

Q. I guess here is my overarching question. I can do the math that looks at a total number of files, divides by 100 and then divides by 3 and I can come to 19. My question is: Is there any math that you can report to or that you can point to either inside your report or anywhere else that is the math that gets me to the calculation that describes how one comes to one day delay per 100 new source code files?

A. So my memory is that we talked about another issue that was reported by Hyperian and it was a real factual situation where the customer reported the problem and the log showed us the time stamps when they addressed it, when they got feedback, what the iterations were, and so, you know, I went through issue 1670 and, you know, basically the report starts on July 5th, 2005 at 8:52 in the morning and at that point he's already used the SAS Learning Edition, then a month or so later Tom Quarendon basically says this isn't important, let's push it out, and we have this log of them pushing it out, and then two and a half years later it's been fixed and, you know, basically there's no more work left to be done. I think there's probably a better example.

- Q. Well, let me start - - let me just take that example. I mean, how could anything that you just described mathematically calculate into one day of delay per 100 files?
- A. Well, in this particular case it appears that it was resolved with minimal effort. It actually appears that it was unintentionally solved just in the normal course of business, so that's not a good example because it shows actually no delay and no effort.

(Hearing Tr., DE 399, 122:18–125:6).

A reliable expert opinion grounded in the expert's experience, training, or education must connect the expert's background to his ultimate conclusion. SAS IV, 2015 WL 4757397, at *7. The more specific the expert's proffered opinion, the more concrete that connection must be. Id. at *9; see also Freeman, 778 F.3d at 468–69 (Agee, J., concurring); Cooper, 259 F.3d at 202. When pressed for a basis for his opinion at his deposition, Roman provided nothing more than a cursory reference to his experience at Dialogic. (See, e.g., Roman Dep. 265:17–267:10). He failed to describe how the experiments conducted by that company informed his conclusion. Moreover, when pressed for the basis of his opinion at the Daubert hearing, Roman repeatedly fell back on publications supporting only the proposition that a lack of subsecond response time occasioned some delay in the software development process. (See, e.g., Hearing Tr. 43:21–44:12, 71:24–73:7). He was unable to identify with specificity any basis in either his experience or academic training to support his opinion that without use of SAS LE development of WPS would have taken an additional 19 days. Roman's total failure to connect his very precise delay calculation with any past source or experience renders his opinion unreliable.

Moreover, as was noted at hearing, assuming that the loss of SAS LE would have created some delay, it simply is too speculative to carry that assumption as far as Roman would take it. An opinion conveying to the jury a precise calculation demands more than an expert's ipse dixit. See

Cooper, 259 F.3d at 203. The specificity of Roman’s 19 day calculation is undercut by his testimony that the delay actually “might be five days, . . . might be 20.” (Hearing Tr. 122:21–22). The danger created by Roman’s cumulative delay calculation, and thus the reason it was excluded, stems from his attempt to punctuate an otherwise imprecise, high-level opinion—that any delay would have been small—with a precise conclusion. While, for example, the more general opinion that lack of a subsecond response time creates some delay might be appropriate under certain circumstances, that was not the issue presented here. Because Roman’s cumulative delay calculation is very specific, in order to allow the court to fully evaluate its reliability, Roman’s experience, training, or academic background supporting his opinion must be similarly specific. That was missing in this case. Thus, the court closed the gate on this testimony.

In any event, at his deposition, Roman also suggested that the number of files that would cause a one day delay also could be explained by research on programmer speed. In effect, he suggested that, because academics have theorized about how quickly programmers can work when creating new source code files, it could be inferred from that research how much delay could be attributed to the average programmer. Roman stated that his experience was informed by the works of Edward Yourdon and The Mythical Man-Month by Frederick Brooks, (see Roman Dep. 278:17–280:25), as well as other cases in which he was a retained expert tasked with measuring programming speed, (Id. 281:1–18), specifically noting that how many new files a programmer could write in one day was “fairly well published.” (Id. 278:147–24). At the Daubert hearing, Roman was questioned extensively about the impact of these publications on his challenged opinion. (Hearing Tr. 70:18–85:16).

In the end, Roman's assumption that every 100 new source code files would have delayed the release of a given version of WPS by one day went unexplained. Although programmer productivity may be well documented, there is no indication that the speed at which a programmer works has any effect on either the amount of time necessary for a given programmer to focus on a new task or the amount of time it takes that same programmer to re-familiarize her or himself with a project after waiting for third parties to provide feedback.

In sum, Roman's cumulative delay opinion lacks a specific and meaningful connection to his experience or sources relied upon. In contrast to Storer's opinion, which the court found relevant and reliable in light of its general nature, see SAS IV, 2015 WL 4757397, at *5-9, Roman's challenged opinion was highly specific. In such a circumstance, where an expert's experiential opinion carries not only the authority inherent in any expert opinion but also an air of precision or exactness, a court should not open the gate unless the expert can connect his calculation, or more generally the methodology used to arrive at that calculation, with a specific instance from his experience, training, or education. Roman could not.

CONCLUSION

For the reasons discussed at hearing, as well as above, plaintiff's motion to exclude Roman's cumulative delay calculation is GRANTED.

SO ORDERED, this the 5th day of November, 2015.

A handwritten signature in black ink, reading "Louise W. Flanagan". The signature is fluid and cursive, with the first name "Louise" being the most prominent part.

LOUISE W. FLANAGAN
United States District Judge